An Integrated Neural Network Model for Domain Action Determination in Goal-Oriented Dialogues

Hyunjung Lee*, Harksoo Kim** and Jungyun Seo***

Abstract—A speaker's intentions can be represented by domain actions (domain-independent speech act and domain-dependent concept sequence pairs). Therefore, it is essential that domain actions be determined when implementing dialogue systems because a dialogue system should determine users' intentions from their utterances and should create counterpart intentions to the users' intentions. In this paper, a neural network model is proposed for classifying a user's domain actions and planning a system's domain actions. An integrated neural network model is proposed for simultaneously determining user and system domain actions using the same framework. The proposed model performed better than previous non-integrated models in an experiment using a goal-oriented dialogue corpus. This result shows that the proposed integration method contributes to improving domain action determination performance.

Keywords—Domain Action, Speech Act, Concept Sequence, Neural Network

1. INTRODUCTION

Dialogue systems are efficient tools for facilitating communication between users and computers using natural languages. They should correctly understand users' utterances and should respond to users' requests, as shown in Table 1.

To realize the former, dialogue systems should identify the underlying intentions of the users' utterances. To realize the latter, dialogue systems should create the counterpart intentions to users' intentions. In other words, when a speaker utters a sentence (a sequence of words), dialogue systems should first classify the uttered sentence into one of the predefined intention categories. This is a process to identify intentions implicated in users' utterances (*i.e.*, a process for catching users' intentions). Then, the dialogue systems should determine their intentions for generating the most suitable sentences in the current dialog history. This process is a process for

---

Corresponding Author: Harksoo Kim
*   Department of Computer Science, Sogang University, Korea (juvenile@sogang.ac.kr)
**   Program of Computer and Communications Engineering, Kangwon National University, Chuncheon-si, Korea (nlpdrkim@kangwon.ac.kr)
***   Department of Computer Science and Interdisciplinary Program of Integrated Biotechnology, Sogang University, Korea (seojy@sogang.ac.kr)

Table 1. An example of utterances along with their corresponding domain actions

Utterance	Domain action
(1) User: Hello.	Greeting & NULL
(2) System: May I help you?	Opening & NULL
(3) User: Tell me tomorrow's schedule.	Request & Timetable-select
(4) System: You have an appointment with Kildong Hong at 11 a.m.	Response & Timetable-select
(5) User: We changed the appointment.	Inform & Timetable-update
(6) System: What was changed?	Ask-ref & Timetable-update
(7) User: The appointment date was changed.	Response & Timetable-update-date
(8) System: What is the new date?	Ask-ref & Timetable-update-date
(9) User: It's December 5.	Response & Timetable-update-date

planning a system's intentions for responding to users' requests. Sentence generation in a goal-oriented dialog can be considered to be a trivial problem, because dialogue systems' responses are restricted to a small set. However, we believe that the selection of a system's intention based on machine learning methods can increase the flexibility of the dialog systems because the priority of the systems' responses is not fixed only by developers but can also be changed by making machine-learned adjustments. As shown in Table 1, specialized speakers' intentions in specific domains can be represented as domain actions consisting of a speech act and concept sequence pairs [1]. Therefore, a domain action determination module should be prepared when developing a dialogue system. However, it is difficult to determine users' domain actions because of their context dependence. For example, the domain action of utterance (9) in Table 1 can be "inform & timetable-select-date" and "response & timetable-update-date" in the surface analysis. To resolve this ambiguity, a dialogue system should analyze the context of utterance (9). In this case, checking the previous utterance (*i.e.,* utterance (8)) is necessary for choosing "response & timetable-update-date" as the domain action of utterance (9). In addition, it is difficult to determine the system's domain actions because they depend on dialogue history and domain knowledge. For example, the dialogue system should consider the dialogue history of, "The appointment date was changed," and the domain knowledge of, "Some necessary information (*e.g.*, location, date, time, and so on) for arranging an appointment" to determine the system's domain action of "ask-ref & timetable-update-date" after analyzing the utterance (7). In this paper, an integrated model is proposed for simultaneously determining user and system domain actions using a neural network framework. In principle, neural networks can compute any computable function (*i.e.*, they can do everything a normal digital computer can do). Anything that can be represented as a mapping between vector spaces can be approximated to arbitrary precision by feed-forward neural networks, which are the most frequently used type. In practice, neural networks are especially useful for solving mapping problems to which hard and fast rules cannot be easily applied. In addition, an advantage of neural networks is that just linking the output nodes of one model with the input nodes of the other model can easily combine independently designed models. Based on this neural network framework advantage, the proposed model combines a speech act determination model with a concept sequence determination model without additional assumptions and labor. The current versions of the proposed models operate in Korean, but language conversion should not be difficult because the models use shallow natural language processing techniques.

   This paper is organized as follows: in Section 2, the previous domain action determination

studies are reviewed. In Section 3, a neural network model is proposed for determining domain action pairs (the user's domain action and the system's domain action) using a dialogue corpus annotated with domain actions as training data. In Section 4, the results of our experiments are analyzed. Finally, some conclusions are drawn in Section 5.

2. RELATED WORK

Previous studies on the determination of users' domain actions are divided into the following two groups: one is the use of handcrafted rules such as recipes for plan inference and domain specific knowledge [2-4], and the other is the use of machine learning techniques based on a large annotated corpus [5-11]. The rule-based methods have shown good performances in small dialogue domains. However, they have problems with scaling-up and changing application domains is very difficult because they depend on costly handcrafted knowledge. Many machine learning methods have recently been proposed for overcoming these problems. The previous machine learning methods have mainly dealt with only speech acts [5, 6, 8, 10, 11] or have independently dealt with speech acts and concept sequences [1, 9]. However, a speech act and concept sequence pair should be simultaneously determined for precisely catching users' intentions because speech acts and concept sequences are tightly associated.

Previous studies on the determination of a system's domain action have been based on dialogue models, such as finite-state models, frame-based models [12], and plan-based models [3, 4]. A finite-state model consists of a set of nodes representing dialogue states and a set of arcs between the nodes. The nodes typically represent the system's responses and the arcs represent user inputs, which move the dialogue from one state to another. The frame-based model uses templates (*i.e.,* collections of information) as the basis of dialogue management. The purpose of the dialogue is to fill necessary information slots in a free order. The frame-based model is more open than the finite-state model because there is no predefined dialogue flow. The plan-based model focuses on the interpretation of utterances and on the construction of plan recipes (*i.e.*, script units that model dialogues). The plan-based model can manage complex dialogue phenomena using plan inferences. However, it is not easy to apply the plan-based model to real world applications because the plan recipes are difficult to maintain.

3. DOMAIN ACTION DETERMINATION USING A NEURAL NETWORK

Given n utterances, $U_{1,n}$, in a dialogue, let $SA_{1,n}$ denote the speech acts of $U_{1,n}$ and $CS_{1,n}$ the concept sequences of $U_{1,n}$. Then, the domain action determination model can be formally defined as the following equation:

$$DA(U_{1,n}) \overset{def}{=} \underset{SA_{1,n},CS_{1,n}}{\arg\max} P(SA_{1,n}, CS_{1,n} \mid U_{1,n}) \tag{1}$$

According to the chain rule, equation (1) is rewritten as equation (2).

$$DA(U_{1,n}) \overset{def}{=} \underset{SA_{1,n},CS_{1,n}}{\arg\max} P(SA_{1,n} \mid CS_{1,n}, U_{1,n}) P(CS_{1,n} \mid U_{1,n}) \tag{2}$$

Equation (2) is then simplified by making the following two assumptions: one is a 1st order Markov assumption that a current category (*i.e.*, a current speech act or a current concept sequence) is dependent on the previous category (*i.e.*, the previous speech act or the previous concept sequence), and the other is a conditional independent assumption that a current category is only dependent on the observational information of the current utterance.

$$DA(U_{1,n}) \approx \arg\max_{SA_{1,n},CS_{1,n}} \prod_{i=1}^{n} P(SA_i \mid CS_i, U_i) P(SA_i \mid SA_{i-1}, CS_{i-1}) P(CS_i \mid U_i) P(CS_i \mid CS_{i-1}) \qquad (3)$$

In equation (3), it is impossible to directly compute $P(SA_i \mid CS_i, U_i)$ and $P(CS_i \mid U_i)$ for the following two reasons:

- If the speaker is a user, he/she will express identical contents using various sentence surface forms according to a personal linguistic sense in a real dialogue.
- If the speaker is a system, surface utterances are realized by their intentions according to cognitive processes. Therefore, the system does not have any surface utterances before intentions are determined.

To overcome these problems, it is assumed that a user's utterance is generalized by a sentential feature set and a system's utterance is realized by a domain-knowledge feature set. Equation (3) is rewritten as equation (4). In equation (4), F_i is a sentential feature set or a domain-knowledge feature set according to its speaker.

$$DA(U_{1,n}) \approx \arg\max_{SA_{1,n},CS_{1,n}} \prod_{i=1}^{n} P(SA_i \mid CS_i, F_i) P(SA_i \mid SA_{i-1}, CS_{i-1}) P(CS_i \mid F_i) P(CS_i \mid CS_{i-1}) \qquad (4)$$

The sentential feature set consists of the following two components: lexical features (content words annotated with POS's) and POS features (POS bi-grams of all the words in an utterance). Generally, content words include nouns, verbs, adjectives, and adverbs, while functional words involve prepositions, conjunctions, and interjections. For example, the sentential feature set of utterance (9) in Table 1 consists of two lexical features ("December/proper-noun," "5/number") and four POS features ("pronoun:verb," "verb:proper-noun," "proper-noun:number," "number:symbol"). In many cases, sentential features are too numerous to be used as inputs for machine learning models. Therefore, methods of removing non-informative features have been required. Yang and Pedersen [13] have performed a comparative study of optimal feature selection for document classification. They have shown that the χ^2 statistic outperforms mutual information and information gain in document classification. Based on Yang and Pedersen's study [13], non-informative sentential features are removed based on the χ^2 statistic. In a goal-oriented dialogue, participants accomplish a given task by using shared domain knowledge.

The domain knowledge can be represented using dialogue models, such as the finite-state model, the frame-based model, and the plan-based model. Since a frame-based model is more flexible than a finite-state model and is easier to implement than a plan-based model, the domain-knowledge feature set is represented based on the frame-based model [14]. The domain-knowledge feature set consists of slot-modification features and slot-retrieval features. The slot-

modification feature represents those slots that are filled with suitable items and the slot-retrieval feature represents those slots that are looked up. The slot-modification feature and the slot-retrieval feature store values in binary notations. For the slot-modification feature, "1" means that the slot is filled with a proper item and "0" means that the slot is empty. For the slot-retrieval feature, a "1" means that the slot has been looked up one or more times. To obtain domain-knowledge features, we predefined domain actions associated with slot modification (*e.g.* "response & timetable-update-date") and slot retrieval (*e.g.* "request & timetable-select-date"), respectively. Then, we automatically generated domain-knowledge features by looking up the predefined domain actions at each dialogue step. In real dialogue systems, dialogue managers will play a role that is similar to looking up the predefined domain actions. Fig. 1 shows an example of the slot-modification and slot-retrieval features.

In Fig. 1, the slot-modification feature value "0 1 0 1 0" indicates that the two slots, "person" and "date," were filled with an item during the dialog from utterance (1) to utterance (11). The slot-retrieval feature value "0 0 0 1 0" indicates that the "date" slot was looked up more than one time during the dialog from utterance (1) to utterance (11). The two kinds of binary values, "0 1 0 1 0" and "0 0 0 1 0," are used as the domain-knowledge feature set F_i in equation (4).

As shown in equation (4), the domain action determination model is formulated as the product of a speech act determination model (the first and the second terms of the right hand side in equation [4]) and a concept sequence determination model (the third and the fourth terms of the right hand side in equation [4]). The speech act determination model and the concept sequence determination model are denoted by SADM and CSDM, respectively. Although SADM and CSDM have different roles, they cannot be dealt with independently because $P(SA_i | CS_i, F_i)$ cannot be computed in SADM if the current concept sequence, CS_i, is not determined in advance. To resolve this problem, a multi-layer perceptron network model is adopted because it
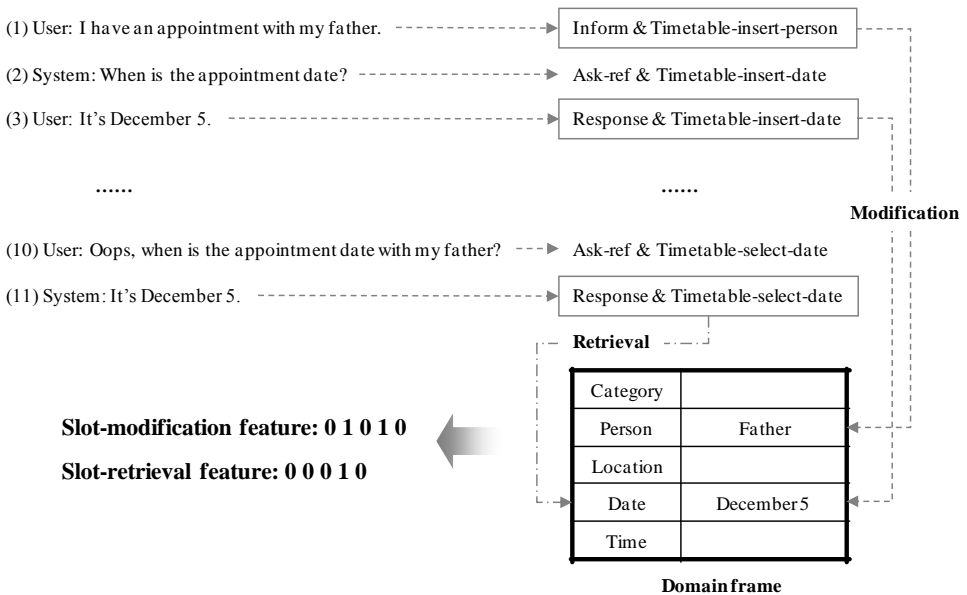

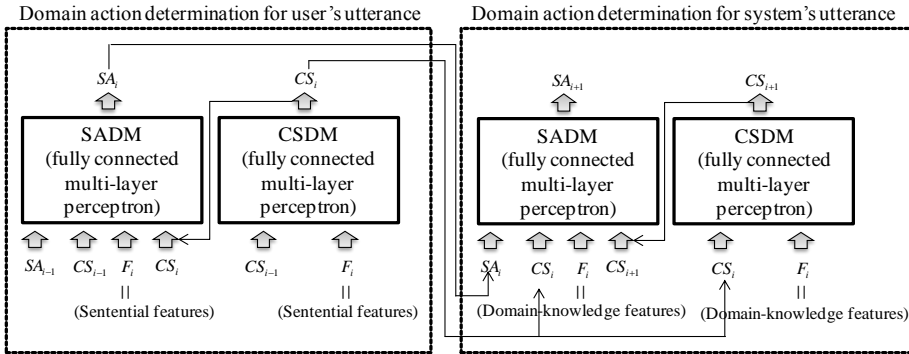Fig. 1. An example of the slot-modification and slot-retrieval features

Fig. 2. Domain action determination model based on a neural network

offers easy and effective methods for combining two associated models. Fig. 2 shows a neural network model for domain action determination based on equation (4).

As shown in Fig. 2, the proposed model combines SADM with CSDM using a neural network framework. SADM and CSDM can be trained at the same time because of the neural network framework. In addition, SADM can naturally use output values (*i.e.*, the possibility distribution of correct concept sequences) of CSDM as input feature values. This framework is expected to increase the robustness of SADM because SADM is less affected by incorrect CSDM outputs.

4. EVALUATION

4.1 Data Sets and Experimental Settings

A Korean dialogue corpus simulated in a schedule management domain, including appointment scheduling and alarm settings, was collected. The dialogue corpus was obtained by eliminating interjections and erroneous expressions from the original transcripts of simulated dialogues between two speakers to whom a dialogue task had been given in advance. This was conducted where one participant freely asks something about his/her daily schedule and the other participant responds to the questions or asks some questions using knowledge bases that were given in advance. This corpus consists of 956 dialogues and 21,336 utterances (22.3 utterances per dialogue). Each dialogue utterance was manually annotated with speech acts and concept sequences. The manual tagging of speech acts and concept sequences was done by 5 graduate students who had knowledge about dialogue analysis. Before manual tagging, we explained the meanings of speech acts and concept sequences to the students and showed them some samples that were annotated with correct speech acts and concept sequences. We spent approximately 2 hours training the students. Then, we assigned one student to each 1/5 of the corpus as a coder. Finally, we post-processed the manually annotated corpus for consistency by a doctoral student. The annotated dialogue corpus was divided into a test corpus with 100 dialogues and a training corpus with 856 dialogues in order to experiment on the proposed model. The training corpus was further divided into 8 parts (100, 200, …, 700, 856 dialogues) to compare the performance with the increasing size. The toolkit used for implementation was SNNS (Stuttgart Neural Network Simulator) [15]. The numbers of input, hidden, and output nodes of the SADM for a user's utterance were 305 (11 for SA_{i-1}, 47 for CS_{i-1}, 200 for F_i,

and 47 for CS_i), 101, and 11 for SA_i, respectively. The numbers of input, hidden, and output nodes of the CSDM for a user's utterance were 257 (47 for CS_{i-1} and 200 for F_i), 82, and 47 for CS_i, respectively. The numbers of input nodes, hidden nodes, and output nodes of the SADM for a system's utterance were 110 (11 for SA_i, 47 for CS_i, 5 for F_i, and 47 for CS_{i+1}), 36, and 11 for SA_{i+1}, respectively. The numbers of input, hidden, and output nodes of the CSDM for a system's utterance were 52 (47 for CS_i and 5 for F_i), 17, and 47 for CS_{i+1}, respectively. We set all of the parameters of the toolkit to default values, except for the learning rate. The learning rate was 0.2, and trainings spent 200 epochs. The proposed model was then evaluated using the accuracy as an evaluation measure, as shown in equation (5).

$$Accuracy = \frac{\#\ of\ correct\ categories}{\#\ of\ returned\ categories} \tag{5}$$

In addition, to statistically validate the performance differences, we performed a Student's *t*-test on each experimental result and have shown the *p*-values obtained by those *t*-tests. As these *p*-values are smaller than 0.01, we can confirm that the performance differences are statistically significant at the 0.01 level.

Core and Allen [16] proposed a standard annotation scheme called DAMSL (Dialog Act Markup in Several Layers). Although DAMSL offers a good annotation scheme, it was not suitable for our experiment domain, which was a schedule management domain, because the number of tag combinations (about 4 million) was too large to annotate simple goal-oriented dialogues. Therefore, a modified subset of the standard DAMSL labels was used in order to reduce the manual annotation labor in the experiment domain, as shown in Table 2. In a goal-oriented dialogue, participants communicate with each other to accomplish a given domain task. Many previous dialogue systems modeled these dialogue behaviors on database operations [9, 14]. To annotate the goal-oriented dialogue corpus according to these previous studies, 47 main actions that map the meanings of utterances into term notations in a database were defined. These included the name of a table, the name of an operation, and the name of a field. For example, the utterance, "The appointment date was changed" is represented as "timetable-update-date."

Table 2. Speech acts and their meanings

Speech act	Description	Example	Percent (%) of corpus
Greeting	The opening greeting of a dialogue	Hello.	9.43
Closing	The closing greeting of a dialogue	Good-bye.	8.79
Opening	Sentences for opening a goal-oriented dialogue	May I help you?	0.02
Ask-ref	WH-questions	Where is the place?	22.78
Ask-if	YN-questions	Can I change the time?	2.76
Response	Responses to questions or requesting actions	Yes, you can.	37.98
Request	Declarative sentences for requesting actions	Set the alarm.	14.24
Ask-confirm	Questions confirming the previous actions	Saturday, right?	0.03
Confirm	Reponses to ask-confirm	Right.	0.03
Inform	Declarative sentences giving some information	It was canceled.	2.07
Accept	Agreement	I know.	1.88

4.2 Experimental Results

The precisions at various cutoff points were calculated in order to evaluate the performances of the proposed model according to various training corpus sizes, as shown in Tables 3 and 4.

In Table 3, the results of Kim et al. [8] are similar to those of the SADM for users' utterances, except that they do not use the concept sequence features as input features. As shown in Table 3, the SADM showed better results than Kim et al. [8] at all of the cut-off points. Moreover, the SADM, which used 100 dialogues as a training corpus, had similar precisions to Kim et al. [8] by using 500-700 dialogues as a training corpus. The *p*-value against Kim et al. [8] is measured as 0.000002. Because the *p*-value is under 0.01, the performance improvement of SADM has a statistical significance. This result shows that the concept sequence features contribute to alleviating the well-known sparse data problem. In addition, it shows that speech acts are tightly associated with concept sequences. The accuracies of CSDM were lower than those of SADM. This may have been caused by the difference between the numbers of target categories, where the target categories of SADM are 11 types of speech acts, but the target categories of CSDM are 53 types of concept sequences.

In Table 4, Reithinger and Klesen [6] is a speech act prediction model (*i.e.*, a speech act planning model) for a system's utterance using *n*-grams of speech acts, as shown in equation (6).

$$P(SA_n \mid SA_{n-1}, SA_{n-2}) = \alpha_1 f(SA_n) + \alpha_2 f(SA_n \mid SA_{n-1}) + \alpha_3 f(SA_n \mid SA_{n-1}, SA_{n-2}) \qquad (6)$$

In equation (6), *f* is the relative frequency of each speech act, *SA*, and α is a weighting factor. In the experiments, α_1, α_2, and α_3 were set to 0.2, 0.3, and 0.5, respectively. Reithinger and Klesen [6] did not perform concept sequence determination evaluations. Therefore, a new model was implemented for concept sequence determination by replacing speech acts with concept sequences in equation (6) and the new model was compared to CSDM. Although there are some previous models for domain action determination [1, 9], we could find few previous models for domain action prediction. Reithinger's model is a representative statistical model that does not need expensive domain knowledge, such as dialogue scripts and sentence patterns. Therefore, we selected Reithinger's model as a comparison model. Seon et al. [17] proposed a domain action prediction model based on a maximum entropy model (MEM) for use as a pre-processor to reduce the search space of an automatic speech recognizer and as a discourse plan-

Table 3. Domain action determination accuracy when a speaker is a user

The size of the training corpus	Speech Act		CSDM
	SADM	Kim et al. [8]	
100	81.09	79.97	66.25
200	78.36	77.96	68.84
300	83.10	79.88	72.33
400	82.34	80.06	71.75
500	84.04	81.85	72.82
600	82.83	81.22	71.93
700	84.00	81.67	73.58
856	86.05	82.79	73.76

Table 4. Domain action determination accuracy when a speaker is a system

The size of the training corpus	Speech Act		Concept Sequence	
	SADM	Reithinger and Klesen [6]	CSDM	Reithinger and Klesen [6]
100	78.76	51.75	79.68	53.63
200	78.23	61.41	81.29	56.13
300	80.02	42.70	82.09	59.27
400	79.21	42.70	82.01	64.73
500	79.30	42.08	82.01	64.19
600	79.75	42.70	81.74	67.14
700	79.30	42.08	82.18	65.71
856	79.84	42.70	81.65	66.97

ner to generate proper responses. Seon's model showed better accuracies (87.25% for speech act prediction, 87.42% for concept sequence prediction) than the proposed model. However, it is difficult to directly compare Seon's model with the proposed model because of the following reasons: first, they used different data as a test set. Second, Seon's model assumed that users' domain actions are always input correctly, because it does not have any domain action determination models for users' utterances. As shown in Table 4, SADM and CSDM outperformed Reithinger and Klesen [6] at all cut-off points. The p-values of SADM and CSDM against Reithinger and Klesen [6] are measured as 0.000002 and 0.000003, respectively. Because the p-values are under 0.01, the performance improvements of SADM and CSDM have a statistical significance. The accuracy differences were mainly caused by input feature diversity. The SADM accuracies for the system's utterances were lower than those of the SADM for users' utterances. This result shows that lexical and POS features are more effective at determining speech acts than domain-knowledge features. The CSDM accuracies for the system's utterances were higher than those of the CSDM for users' utterances. This may have been caused by the simplicity of the schedule management domain in which the concept sequences of the next utterances are greatly affected by the current domain frame information because the main roles of dialogue participants are to fill empty slots by asking for missing information. Based on this experiment, it has been concluded that domain-knowledge features give good clues for the concept sequence determination for a system's utterances in simple goal-oriented dialogues.

4.3 Failure Analysis

The cases in which the proposed model failed to return correct results were analyzed. The reasons for failure are as given below.

- Well-known context errors: the proposed model used a linearly adjacent speech act (or a linearly adjacent concept sequence) as contextual information. However, dialogues have a hierarchical discourse structure. In the following example, knowing the utterance domain action (4), utterance (3), and not utterance (1) should be considered contextual information because utterance (4) is adjacent to utterance (1) in the tree structure of the discourse. To overcome this problem, methods of applying discourse structures to the proposed model should be studied.

(1) User: "I'd like to change the appointment time."
(2) System: "To what time do you want to change it?"
(3) User: "4 p.m."
(4) System: "I changed it."

• Independence errors: Concept sequences were assumed to be independent of speech acts. However, it was discovered that previous concept sequences, as well as previous speech acts, should sometimes be considered in order to determine correct speech acts. In the following example, utterance (3) has several surface speech acts, such as *inform* and *response*.

(1) System: "What was changed?"
(2) User: "The appointment date was changed."
(3) User: "It is December 5."

Such an ambiguity can be solved by considering the concept sequence of utterance (3). If only the speech act of utterance (2) is considered, the speech act of utterance (3) may be *inform*. However, if the concept sequence of utterance (3) is considered, utterance (3) can be found to be closely associated with utterance (2) and its meaning is the changed date. Based on these facts, the speech act of utterance (3) must be *response* for the user to provide the system with additional information (*cf*. "The appointment date was changed to December 5"). To overcome this problem, linguistic relations between speech acts and concept sequences should be studied.

5. CONCLUSION

An an integrated neural network model was proposed for simultaneously determining user and system domain actions. The proposed model divides a domain action into a speech act and a concept sequence, which are tightly associated with each other. The proposed model easily combines the speech act determination model and the concept sequence determination model based on the neural network framework without additional assumptions and labor. Moreover, the proposed model can determine users' domain actions and a system's domain action without any changes in the model's architecture occurring. The proposed model performed better than the previous non-integrated models in an experiment with a goal-oriented dialogue corpus, which shows that the proposed integration method is effective at determining domain actions.

REFERENCES

[1]   L. Levin, C. Langley, A. Lavie, D. Gates, D. Wallace and Peterson, K., "*Domain Specific Speech Acts for Spoken Language Translation,*" *Proceedings of 4th SIGdial Workshop on Discourse and Dialogue*, 2003.

[2]   S. Carberry, "A Pragmatics-Based Approach to Ellipsis Resolution," *Computational Linguistics*, vol.15, no.2, 1989, pp.75-96.

[3]   L. Lambert and S. Carberry, "*A Tripartite Plan-Based Model of Dialogue,*" *Proceedings of ACL*, 1991, pp.47-54.

[4]   D. J. Litman and J. F. Allen, "A Plan Recognition Model for Subdialogues in Conversations," *Cogni-*

tive Science, Vol.11, 1987, pp.163-200.

[5] M. Nagata and T. Morimoto, "First Steps towards Statistical Modeling of Dialogue to Predict the Speech Act Type of the Next Utterance," *Speech Communication*, Vol.15, 1994, pp.193-203.

[6] N. Reithinger and M. Klesen, "*Dialogue Act Classification Using Language Models," Proceedings of EuroSpeech*, 1997, pp.2235-2238.

[7] K. Samuel, S. Carberry and K. Vijay-Shanker, "*Computing Dialogue Acts from Features with Trans-form-based Learning," Proceedings of AAAI Spring Symposium*, 1998, pp.90-97.

[8] K. Kim, H. Kim and J. Seo, "A Neural Network Model with Feature Selection for Korean Speech Act Classification," *International Journal of Neural Systems*, Vol.14, No.6, 2004, pp.407-414.

[9] H. Lee, H. Kim and J. Seo, "Domain Action Classification Using a Maximum Entropy Model in a Schedule Management Domain," *AI Communications*, Vol.21, No.4, 2008, pp.221-229.

[10] S. Kang, H. Kim and J. Seo, "A Reliable Multidomain Model for Speech Act Classification," *Pattern Recognition Letters*, Vol.31, No.1, 2010, pp.71-74.

[11] H. Kim, C.-N. Seon, and J. Seo, "Review of Korean Speech Act Classification: Machine Learning Methods," *Journal of Computing Science and Engineering*, Vol.5, No.4, 2011, pp.288-293.

[12] D. Goddeau, H. Meng, J. Polifroni, S. Seneff and S. Busayapongchai, "*A Form-Based Dialogue Manager for Spoken Language Applications," Proceedings of International Conference on Spoken Language Processing*, 1996, pp.701-704.

[13] Y. Yang and J. O. Pedersen, "*A Comparative Study on Feature Selection in Text Categorization," Proceedings of the 14th International conference on Machine Learning*, 1997.

[14] D. Kim, H. Lee, C.-N. Seon, H. Kim and J. Seo, "*Speakers' Intention Prediction using Statistics of Multi-Level Features in a Schedule Management Domain," Proceedings of ACL-08: HLT*, 2008, pp.229-232.

[15] SNNS (Stuttgart Neural Network Simulator), http://www.ra.cs.uni-tuebingen.de/SNNS

[16] M. Core and J. Allen, "*Coding Dialogues with the DAMSL Annotation Scheme," Proceedings of AAAI Fall Symposium on Communicative Action in Humans and Machines*, 1997.

[17] C.-N. Seon, H. Kim and J. Seo, "A Statistical Prediction Model of Speakers' Intentions Using Multi-level Features in a Goal-oriented Dialogue System," *Pattern Recognition Letters*, Vol.33, No.6, 2012, pp.1397-1404.

Hyunjung Lee

She is a Ph.D. student majoring in Natural Language Processing at Sogang University. She received a BA degree in computer science from Dongduk Women's University in 1995 and a MS degree in computer science from Sogang University in 1997. She has worked for NHN corp. as a principal researcher from 2000. Her research interests include natural language processing, dialogue understanding and information retrieval.

Harksoo Kim

He is an associate professor of Computer and Communications Engineering at Kangwon National University. He received his BA in Computer Science from Konkuk University in 1996. He received his MS and PhD in Computer Science from Sogang University in 1998 and 2003, respectively. He visited the CIIR at the University of Massachusetts in Amherst as a research fellow in 2004. In 2005, he worked for the Electronics and Telecommunications Research Institute (ETRI) as a senior researcher. His research interests include natural language processing, dialogue system, information retrieval, and question-answering.

Jungyun Seo

He is a full professor of computer science at Sogang University. He was educated at Sogang University, where he obtained a BS degree in mathematics in 1981. He continued his studies at the department of computer science in the University of Texas, Austin, receiving a MS and Ph.D. in computer science in 1985 and 1990 respectively. He returned to Korea in 1991 to join the faculty of Korea Advanced Institute of Science and Technology (KAIST) in Taejon where he leaded the Natural Language Processing Laboratory in Computer Science Department. In 1995, he moved to the Sogang University in Seoul and became a full professor in 2001. His research interests include multi-modal dialogues, statistical methods for NLP, machine translation and information retrieval.